

**PENDING CLAIMS**

1. (Previously Presented): A method in a data processing system for specifying a cache policy for caching pages which include dynamic content, said method comprising the steps of:
  - permitting a user to request one of said pages to be displayed, said one of said pages including a plurality of fragments;
  - executing an application which includes a plurality of servlets, each one of said plurality of servlets being executed to present a different one of said plurality of fragments, each one of said plurality of servlets being unchanged by said caching policy;
  - processing caching of said one of said pages separately from said application; and
  - specifying one of a plurality of different caching options for one of said plurality of servlets based on an update rate of content of said one of said plurality of servlets, said plurality of different caching options including either static caching, dynamic caching or no caching, content that is updated dynamically being cached using either static caching, dynamic caching, or no caching.
2. (Original): The method according to claim 1, further comprising the steps of processing caching of each of said plurality of fragments separately from said application.
3. (Original): The method according to claim 1, wherein the step of processing caching further comprises the steps of:
  - responding to internal cache requests; and
  - responding to external cache requests.
4. (Original): The method according to claim 1, further comprising the step of processing caching of said one of said pages within an application server included within said computer system.

5. (Original): The method according to claim 4, further comprising the steps of:  
    responding to internal cache requests, said internal cache requests being generated within said application server; and  
    responding to external cache requests, said external cache requests being generated outside said application server.
6. (Original): The method according to claim 1, wherein the step of processing caching of said one of said pages further comprises the steps of:  
    receiving a request to display one of said plurality of fragments;  
    determining one of plurality of servlets associated with said one of said plurality of fragments; and  
    executing said one of said plurality of servlets, wherein said execution of said one of said plurality of servlets generates a displayable output.
7. (Original): The method according to claim 6, further comprising the steps of:  
    in response to a first request to display said one of said plurality of fragments, creating a cache entry including said output;  
    creating a cache entry identifier for identifying said cache entry utilizing a user identifier which identifies said user and caching options specified for said one of said plurality of servlets.
8. (Original): The method according to claim 7, further comprising the steps of:  
    creating said one of said plurality of servlets;  
    specifying said cache options for said one of said plurality of servlets; and  
    creating a servlet element for said servlet utilizing a servlet identifier and an indication of said specified cache options, wherein said servlet element is associated with said servlet.
9. (Original): The method according to claim 8, further comprising the steps of:  
    storing said servlet; and  
    storing said specification of said servlet options with said servlet.

10. (Original): The method according to claim 8, further comprising the steps of:  
receiving a request to display said servlet element;  
determining whether any cache entry is identified by said cache identifier;  
in response to a determination that no cache entry is identified by said cache identifier:  
retrieving said servlet associated with said servlet element;  
providing said user identifier to said servlet;  
executing said servlet utilizing said user identifier generating an output;  
storing said output as a cache entry;  
identifying said cache entry utilizing said cache identifier; and  
returning said cache entry to said user, wherein said output is displayed.
11. (Original): The method according to claim 8, further comprising the steps of:  
receiving a request to display said servlet element;  
determining whether any cache entry is identified by said cache identifier;  
in response to a determination that a cache entry exists which is identified by said cache identifier, returning said cache entry to said user, wherein said output is displayed.
12. (Original): The method according to claim 6, further comprising the step of  
outputting said cache entry, wherein said one of said plurality of fragments is displayed.
13. (Original): The method according to claim 6, further comprising the step of in  
response to subsequent requests to display said one of said plurality of fragments,  
retrieving said cache entry utilizing said cache identifier.
14. (Original): The method according to claim 13, further comprising the step of  
outputting said cache entry, wherein said one of said plurality of fragments is displayed.
15. (Previously Presented): A data processing system for specifying a cache policy  
for caching pages which include dynamic content, comprising:

said data processing system for executing an application which includes a plurality of servlets, each one of said plurality of servlets being executed to present a different one of a plurality of fragments included within a page, each one of said plurality of servlets being unchanged by said caching policy;

said data processing system for processing caching of said one of said pages separately from said application; and

one of a plurality of different caching options specified for one of said plurality of servlets based on an update rate of content of said one of said plurality of servlets, said plurality of different caching options including either static caching, dynamic caching or no caching, content that is updated dynamically being cached using either static caching, dynamic caching, or no caching.

16. (Original): The system according to claim 15, further comprising said data processing system for processing caching of each of said plurality of fragments separately from said application.

17. (Original): The system according to claim 15, further comprising:  
said data processing system for responding to internal cache requests; and  
said data processing system for responding to external cache requests.

18. (Original): The system according to claim 15, further comprising said data processing system for processing caching of said one of said pages within an application server included within said computer system.

19. (Original): The system according to claim 18, further comprising:  
said data processing system for responding to internal cache requests, said internal cache requests being generated within said application server; and  
said data processing system for responding to external cache requests, said external cache requests being generated outside said application server.

20. (Original): The system according to claim 15, further comprising:  
said data processing system for receiving a request to display one of said plurality of fragments;  
said data processing system for determining one of plurality of servlets associated with said one of said plurality of fragments; and  
said data processing system for executing said one of said plurality of servlets, wherein said execution of said one of said plurality of servlets generates a displayable output.
21. (Original): The system according to claim 20, further comprising:  
said data processing system for in response to a first request to display said one of said plurality of fragments, creating a cache entry including said output;  
said data processing system for creating a cache entry identifier for identifying said cache entry utilizing a user identifier which identifies said user and caching options specified for said one of said plurality of servlets.
22. (Original): The system according to claim 21, further comprising:  
said data processing system for creating said one of said plurality of servlets;  
said data processing system for specifying said cache options for said one of said plurality of servlets; and  
said data processing system for creating a servlet element for said servlet utilizing a servlet identifier and an indication of said specified cache options, wherein said servlet element is associated with said servlet.
23. (Original): The system according to claim 22, further comprising:  
said data processing system for storing said servlet; and  
said data processing system for storing said specification of said servlet options with said servlet.
24. (Original): The system according to claim 22, further comprising:  
said data processing system for receiving a request to display said servlet element;

said data processing system for determining whether any cache entry is identified by said cache identifier;

said data processing system in response to a determination that no cache entry is identified by said cache identifier:

for retrieving said servlet associated with said servlet element;

for providing said user identifier to said servlet;

for executing said servlet utilizing said user identifier generating an output;

for storing said output as a cache entry;

for identifying said cache entry utilizing said cache identifier; and

for returning said cache entry to said user, wherein said output is displayed.

25. (Original): The system according to claim 22, further comprising:  
said data processing system for receiving a request to display said servlet element;  
determining whether any cache entry is identified by said cache identifier;  
said data processing system for in response to a determination that a cache entry exists which is identified by said cache identifier, returning said cache entry to said user, wherein said output is displayed.

26. (Original): The system according to claim 20, further comprising said data processing system for outputting said cache entry, wherein said one of said plurality of fragments is displayed.

27. (Original): The system according to claim 20, further comprising said data processing system for in response to subsequent requests to display said one of said plurality of fragments, retrieving said cache entry utilizing said cache identifier.

28. (Original): The system according to claim 27, further comprising said data processing system for outputting said cache entry, wherein said one of said plurality of fragments is displayed.

29. (Previously Presented): A computer program product in a data processing system for specifying a cache policy for caching pages which include dynamic content, said computer program product comprising:

instruction means for permitting a user to request one of said pages to be displayed, said one of said pages including a plurality of fragments;

instruction means for executing an application which includes a plurality of servlets, each one of said plurality of servlets being executed to present a different one of said plurality of fragments, each one of said plurality of servlets being unchanged by said caching policy;

instruction means for processing caching of said one of said pages separately from said application; and

instructions for specifying one of a plurality of different caching options for one of said plurality of servlets based on an update rate of content of said one of said plurality of servlets, said plurality of different caching options including either static caching, dynamic caching or no caching, content that is updated dynamically being cached using either static caching, dynamic caching, or no caching.

30. (Original): The product according to claim 29, further comprising instruction means for processing caching of each of said plurality of fragments separately from said application.

31. (Original): The product according to claim 29, wherein said instruction means for processing caching further comprises:

instruction means for responding to internal cache requests; and

instruction means for responding to external cache requests.

32. (Original): The product according to claim 29, further comprising instruction means for processing caching of said one of said pages within an application server included within said computer system.

33. (Original): The product according to claim 32, further comprising:  
instruction means for responding to internal cache requests, said internal cache requests being generated within said application server; and  
instruction means for responding to external cache requests, said external cache requests being generated outside said application server.
34. (Original): The product according to claim 29, wherein said instruction means for processing caching of said one of said pages further comprises:  
instruction means for receiving a request to display one of said plurality of fragments;  
instruction means for determining one of plurality of servlets associated with said one of said plurality of fragments; and  
instruction means for executing said one of said plurality of servlets, wherein said execution of said one of said plurality of servlets generates a displayable output.
35. (Original): The product according to claim 34, further comprising:  
instruction means for in response to a first request to display said one of said plurality of fragments, creating a cache entry including said output;  
instruction means for creating a cache entry identifier for identifying said cache entry utilizing a user identifier which identifies said user and caching options specified for said one of said plurality of servlets.
36. (Original): The product according to claim 35, further comprising:  
instruction means for creating said one of said plurality of servlets;  
instruction means for specifying said cache options for said one of said plurality of servlets; and  
instruction means for creating a servlet element for said servlet utilizing a servlet identifier and an indication of said specified cache options, wherein said servlet element is associated with said servlet.



37. (Original): The product according to claim 36, further comprising:  
instruction means for storing said servlet; and  
instruction means for storing said specification of said servlet options with said servlet.
38. (Original): The product according to claim 36, further comprising:  
instruction means for receiving a request to display said servlet element;  
instruction means for determining whether any cache entry is identified by said cache identifier;  
instruction means in response to a determination that no cache entry is identified by said cache identifier:  
for retrieving said servlet associated with said servlet element;  
for providing said user identifier to said servlet;  
for executing said servlet utilizing said user identifier generating an output;  
for storing said output as a cache entry;  
for identifying said cache entry utilizing said cache identifier; and  
for returning said cache entry to said user, wherein said output is displayed.
39. (Original): The product according to claim 36, further comprising:  
instruction means for receiving a request to display said servlet element;  
instruction means for determining whether any cache entry is identified by said cache identifier;  
instruction means for in response to a determination that a cache entry exists which is identified by said cache identifier, returning said cache entry to said user, wherein said output is displayed.
40. (Original): The product according to claim 34, further comprising instruction means for outputting said cache entry, wherein said one of said plurality of fragments is displayed.

41. (Original): The product according to claim 34, further comprising instruction means for in response to subsequent requests to display said one of said plurality of fragments, retrieving said cache entry utilizing said cache identifier.

42. (Original): The product according to claim 41, further comprising instruction means for outputting said cache entry, wherein said one of said plurality of fragments is displayed.